Modeling

Aleksei Sorokin, asorokin@hawk.iit.edu, A20394300

3/12/2020

Load Data

```
df <- read.csv('data/all_coaches.csv')
head(df)
```

```
##              N       GR       WL.       PGR  PWL. CC C HOF    Sport
## 1     AJ Hinch 6.308642 0.5580000 4.5454545 0.560  2 1   0 baseball
## 2  Aaron Boone 2.000000 0.6270000 1.2727273 0.500  0 0   0 baseball
## 3 Aaron Kromer 0.375000 0.3330000 0.0000000 0.000  0 0   0 football
## 4   Abe Gibron 2.625000 0.2740000 0.0000000 0.000  0 0   0 football
## 5    Adam Gase 4.000000 0.4690000 0.3333333 0.000  0 0   0 football
## 6   Adam Oates 1.585366 0.5752212 0.4375000 0.429  0 0   0   hockey
```

Split into train test datasets

```
set.seed(7)
train_frac <- 4/5
df_HOF_1 <- df[df$HOF==1,]
df_HOF_0 <- df[df$HOF==0,]
HOF_1_train <- sample(1:nrow(df_HOF_1),floor(nrow(df_HOF_1)*train_frac))
HOF_0_train <- sample(1:nrow(df_HOF_0),floor(nrow(df_HOF_0)*train_frac))
df_train <- rbind(df_HOF_1[HOF_1_train,],df_HOF_0[HOF_0_train,])
df_test <- rbind(df_HOF_1[-HOF_1_train,],df_HOF_0[-HOF_0_train,])
l1 <- sprintf('Train Fractio:n %.2f',train_frac)
l2 <- sprintf('Hall of Fame Coaches: %d. (Train %d , Test %d)',
       nrow(df_HOF_1),length(HOF_1_train),nrow(df_HOF_1)-length(HOF_1_train))
l3 <- sprintf('Non Hall of Fame Coaches: %d. (Train %d , Test %d)',
       nrow(df_HOF_0),length(HOF_0_train),nrow(df_HOF_0)-length(HOF_0_train))
l4 <- sprintf('Overall: (Train %d , Test %d)',nrow(df_train),nrow(df_test))
cat(sprintf('%s\n%s\n%s\n%s\n',l1,l2,l3,l4))
```

```
## Train Fractio:n 0.80
## Hall of Fame Coaches: 256. (Train 204 , Test 52)
## Non Hall of Fame Coaches: 1664. (Train 1331 , Test 333)
## Overall: (Train 1535 , Test 385)
```

Standard Logistic Regression Model

```r
model_1 <- glm(HOF ~ GR + PGR + CC + Sport,#GR+WL.+PGR+PWL.+CC+C+Sport,
               data=df_train,
               family="binomial")
summary(model_1)
```

```
##
## Call:
## glm(formula = HOF ~ GR + PGR + CC + Sport, family = "binomial",
##     data = df_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0454  -0.4715  -0.4047  -0.2774   2.6366
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -2.62049    0.17797 -14.724  < 2e-16 ***
## GR               0.05246    0.02509   2.091   0.0365 *
## PGR             -0.14789    0.06572  -2.250   0.0244 *
## CC               0.87780    0.11523   7.618 2.58e-14 ***
## Sportbasketball -0.76418    0.34003  -2.247   0.0246 *
## Sportfootball    0.15601    0.22063   0.707   0.4795
## Sporthockey      1.01268    0.22845   4.433 9.30e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1203.01  on 1534  degrees of freedom
## Residual deviance:  996.91  on 1528  degrees of freedom
## AIC: 1010.9
##
## Number of Fisher Scoring iterations: 5
```

```r
df_test$yHat_1 <- predict(model_1, newdata=df_test, type="response")
```

Logistic Mixed Model

```r
library(lme4)
model_2 <- lmer(HOF ~ GR + PGR + CC + (1|Sport),#GR + WL. + PGR + PWL. + CC + C + (1 | Sport),
                data = df_train)
summary(model_2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: HOF ~ GR + PGR + CC + (1 | Sport)
##    Data: df_train
##
## REML criterion at convergence: 786.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.4952 -0.4318 -0.2593 -0.1288  3.1962
##
```

```
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Sport    (Intercept) 0.004102 0.06405
##  Residual             0.095276 0.30867
## Number of obs: 1535, groups:  Sport, 4
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  0.082629   0.033673   2.454
## GR           0.006294   0.002490   2.528
## PGR         -0.017751   0.006540  -2.714
## CC           0.122029   0.009882  12.349
##
## Correlation of Fixed Effects:
##     (Intr) GR     PGR
## GR  -0.151
## PGR  0.019 -0.610
## CC   0.048 -0.243 -0.381
```

```
df_test$yHat_2 <- predict(model_2,df_test, type="response")
```

Bayesian Logistic Mixed Model

```
library(blme)
model_3 <- blmer(HOF ~ GR + PGR + CC + (1|Sport),# GR + WL. + PGR + PWL. + CC + C + (1 | Sport),
                 data = df_train,
                 resid.prior = gamma,
                 fixef.prior = normal,
                 cov.prior = wishart)
summary(model_3)
```

```
## Cov prior  : Sport ~ wishart(df = 3.5, scale = Inf, posterior.scale = cov, common.scale = TRUE)
## Fixef prior: normal(sd = c(10, 2.5, ...), corr = c(0 ...), common.scale = TRUE)
## Resid prior: gamma(shape = 0, rate = 0, posterior.scale = var)
## Prior dev  : 6.8976
##
## Linear mixed model fit by REML ['blmerMod']
## Formula: HOF ~ GR + PGR + CC + (1 | Sport)
##    Data: df_train
##
## REML criterion at convergence: 787.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5022 -0.4277 -0.2591 -0.1218  3.2136
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Sport    (Intercept) 0.00888  0.09424
##  Residual             0.09481  0.30791
## Number of obs: 1535, groups:  Sport, 4
##
## Fixed effects:
```

```
##              Estimate Std. Error t value
## (Intercept)  0.082544   0.048245   1.711
## GR           0.006355   0.002485   2.557
## PGR         -0.017970   0.006538  -2.748
## CC           0.122184   0.009863  12.389
##
## Correlation of Fixed Effects:
##     (Intr) GR     PGR
## GR  -0.105
## PGR  0.013 -0.610
## CC   0.034 -0.242 -0.383
```

```
df_test$yHat_3 <- predict(model_3, newdata=df_test, type="response")
```

Make predictions on test data and calculate metrics

```
model_metrics <- function(df_test,yHat_col,model_name){
  yHat_b_col <- paste0(yHat_col,'b')
  df_test[,yHat_b_col] <-(df_test[[yHat_col]] >= .5)
  yHat_b <- df_test[[yHat_b_col]]
  hof <- df_test$HOF
  tp <- sum((yHat_b==1 & hof==1))
  fp <- sum((yHat_b==1 & hof==0))
  fn <- sum((yHat_b==0 & hof==1))
  tn <- sum((yHat_b==0 & hof==0))
  accuracy <- (tp+tn)/(tp+tn+fp+fn)
  precision <- tp/(tp+fp)
  recall <- tp/(tp+fn)
  cat(sprintf('%s\n\tAccuracy: %.3f\n\tPrecision: %.3f\n\tRecall: %.3f\n',
              model_name,accuracy,precision,recall))
  return (df_test)}
```

Compare models

```
df_test <- model_metrics(df_test,'yHat_1','Standard Logistic Regression')
```

```
## Standard Logistic Regression
##  Accuracy: 0.888
##  Precision: 0.765
##  Recall: 0.250
```

```
df_test <- model_metrics(df_test,'yHat_2','Logistic Mixed Model')
```

```
## Logistic Mixed Model
##  Accuracy: 0.891
##  Precision: 1.000
##  Recall: 0.192
```

```
df_test <- model_metrics(df_test,'yHat_3','Bayesian Mixed Model')
```

```
## Bayesian Mixed Model
##  Accuracy: 0.891
##  Precision: 1.000
```

```
##  Recall: 0.192
```

```
head(df_test[df_test$HOF==1,])
```

```
##                   N        GR       WL.      PGR  PWL. CC C HOF      Sport
## 31   Alan Trammell  3.018519 0.3820000 0.000000 0.000  0 0   1   baseball
## 39     Alex Hannum 10.768293 0.5330000 4.937500 0.570  2 2   1 basketball
## 113      Bill Cook  1.426829 0.3655914 0.000000 0.000  0 0   1     hockey
## 114    Bill Cowher 15.000000 0.6230000 7.000000 0.571  2 1   1   football
## 135 Bill McKechnie 22.512346 0.5240000 2.000000 0.364  4 2   1   baseball
## 156     Bill Terry  9.234568 0.5550000 1.454545 0.438  3 1   1   baseball
##         yHat_1     yHat_2    yHat_3 yHat_1b yHat_2b yHat_3b
## 31  0.07855474 0.08314347 0.0827151   FALSE   FALSE   FALSE
## 39  0.14253377 0.24481337 0.2418411   FALSE   FALSE   FALSE
## 113 0.17756169 0.17706062 0.1806096   FALSE   FALSE   FALSE
## 114 0.27742644 0.29189626 0.2913072   FALSE   FALSE   FALSE
## 135 0.85518225 0.65846170 0.6593888    TRUE    TRUE    TRUE
## 156 0.57008846 0.46253822 0.4626313    TRUE   FALSE   FALSE
```